

2725 in the amount of \$120.00 for the extension fee. Please consider this a Petition for Extension of Time for a sufficient number of months to enter this correspondence, or any future reply, if appropriate, for an extension of time for its timely submission.

At any time during the pendency of this application, please charge all fees required under 37 CFR §§ 1.16, 1.17, and 1.18, or credit any overpayment to, Deposit Account 13-2725 pursuant to 37 CFR § 1.25(b).

This paper is also in response to a telephone interview with the Examiner that was held on January 29, 2008. A brief summary of the telephone interview is provided below.

In response to the Office Action, Part of Paper No./Mail Date 20071010, mailed on October 17, 2007, please amend the above identified application as follows:

Amendments To The Claims are reflected in the listing of claims which begins on page 3 of this paper.

Remarks/Arguments begin on page 6 of this paper.

AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listing, of claims in the application:

Listing of Claims:

1. **(Currently Amended)** In a computer system having an operating system platform, a user interface framework system for rendering data according to a visual style defined for the data type, the system comprising:

a plurality of data objects, wherein the plurality of data objects have one or more data fields;

a style definition module for holding one or more visual style definitions to be selectively applied to the one or more data fields of the plurality of data objects;

a style lookup module for locating an associated visual style definition;

a binding module for binding the one or more data fields to one of a properties of the ~~appropriate~~associated visual style definition;

a tree assembler module for generating a visual tree which is an internal representation of a combination of the one or more data fields and based on the ~~appropriate~~associated visual style definition;

a user interface element factory having additional user interface elements;

a layout engine for adding one or more of the additional user interface elements to the visual tree after the ~~appropriate~~associated visual style definition has been bound to the one or more data fields, wherein the layout engine is able to place said additional user interface elements relative to the physical properties of a display of the computer system; and

a rendering engine which uses the visual tree passed to the layout engine which added the one or more user interface elements to render the data for display.

2. **(Cancelled)**

3. **(Currently Amended)** A ~~The~~ system as defined in claim 1 wherein the plurality of data objects are independent from the one or more visual style definitions~~styles~~.

4. **(Canceled)**

2 5. **(Currently Amended)** ~~A~~ The system as defined in claim 1 wherein the plurality of
2 data objects are displayed as a list.

2 6. **(Currently Amended)** ~~A~~ The system as defined in claim 1 wherein the plurality of
2 data objects are displayed as a menu.

2 7. **(Currently Amended)** ~~A~~ The system as defined in claim 1 wherein the plurality of
2 data objects are displayed as a combo box.

2 8. **(Currently Amended)** ~~A~~ The system as defined in claim 1 wherein the plurality of
2 data objects form a group, and wherein the system further comprises a group visual style
4 definition and wherein the tree assembler module generates the visual ~~representation-tree~~ based
4 on the group visual style, the group visual style being independently defined from the ~~data-items~~.

2 9. **(Currently Amended)** A method of displaying one or more data items according to
2 an ~~appropriate~~ associated visual style comprising:

4 receiving a request to display one or more data items;

4 locating the ~~appropriate~~ associated visual style, wherein the ~~appropriate~~ associated visual
style is independently defined from the one or more data items;

6 generating a visual tree which is an internal representation of a combination of using the
one or more data items and the ~~appropriate~~ associated visual style;

8 binding properties of the associated visual style in the visual tree to properties of the one
or more data items;

10 adding additional user interface elements to the visual tree after the properties of the
associated visual style in the visual tree have been bound to the one or more data items, wherein
12 the additional user interface elements are placed relative to the physical properties of a display;
and

14 rendering the one or more data items for display based on the visual tree having the
additional user interface elements.

2 10. **(Currently Amended)** ~~A~~ The method as defined in claim 9 further comprising
2 declaring the one or more data items using data objects.

11. **(Currently Amended)** ~~A-The~~ method as defined in claim 9 further comprising:
automatically updating the visual tree in response to a change to a relevant data item.[.]

12. **(Currently Amended)** ~~A-The~~ method as defined in claim 11 wherein the change to a
relevant data item involves the addition of a relevant data item.

13. **(Currently Amended)** ~~A-The~~ method as defined in claim 11 wherein the change to a
relevant data item involves the deletion of a relevant data item.

14. **(Cancelled)**

15. **(Currently Amended)** ~~A-The~~ method as defined in claim 9 wherein the one or more
data items form a list.

16. **(Currently Amended)** ~~A-The~~ method as defined in claim 9 wherein the one or more
data items form a menu.

17. **(Currently Amended)** ~~A-The~~ method as defined in claim 9 wherein the one or more
data items form a ~~combination~~combo box.

18. **(Currently Amended)** ~~A-The~~ method as defined in claim 9 further comprising:
defining a visual style for a group;
associating the one or more data items with the group;
in response to the request to display the one or more data items, locating the visual style
for the group; and
generating the visual tree based on the visual style for the group.

19. **(Currently Amended)** ~~A-The~~ method as defined in claim 9 further comprising:
detecting a change dynamically in a relevant data item;
invalidating the visual tree;
recognizing the invalidation of the visual tree;
in response to recognizing the invalidation of the visual tree, regenerating the necessary
portions of the visual tree; and
re-rendering the display based on the regenerated visual tree.